Remarks/Arguments

Claims 1-27 were pending in the final Office Action dated October 18, 2011. Claims 1-27 are canceled herein, and Claims 28-42 are new. Is it respectfully submitted that no new matter has been added.

Claims 1-27 were rejected under 35 USC § 103(a) as being unpatentable over Leupers *et al.*, "Generation of Interpretive and Compiled Instruction Set Simulators" ("Leupers"), U.S. Patent No. 5,781,758 to Morley ("Morley"), U.S. Patent No. 7,107,580 to Zemach *et al.* ("Zemach"), U.S. Pub. No. 2005/0102493 by DeWitt, Jr. *et al* ("DeWitt"), U.S. Patent No. 6,477,683 to Killian *et al.* ("Killian"), U.S. Pub. No. 2003/0217248 by Nohl *et al.* ("Nohl"), and U.S. Pub. No. 2005/0160402 by Wang *et al.* ("Wang").

I. Interview Summary

Applicant sincerely appreciates the Examiner's time for a phone conversation to discuss the patentability of the pending claims on January 31, 2012. In the interview, the Examiner and Applicant agreed that the differences between the disclosed invention and the cited prior art are not reflected in the claims as currently recited. In the newly presented claims and the following sections of this paper, Applicant submits responses based on the Examiner's interview.

II. Rejections under 35 USC §103

A. Leupers, Morley, and Zemach.

Applicants respectfully submit that new Claims 28-42 are patentable under 35 U.S.C. §103(a) over Leupers in view of Morley, further in view of Zemach for at least the following reasons.

Independent Claims 28, 32, and 40 each recite:

> **generating decoder source code,** prior to a simulation time, by ...identifying a plurality of unique instruction patterns across a plurality of input applications; selecting an appropriate template... configured to implement a functionality of an instruction contained within an instruction class...generating a customized template for each instruction of the plurality of instructions using parameter values contained in the

> instruction…**compiling the decoder source code to generate an optimized decoder having optimized decoded instructions**.

Claims 28, 32, and 40, emphasis added. As the Examiner acknowledged, Leupers does not provide such a teaching. 10/18/2011 Office Action, p. 3.

Morley does not provide such a teaching either. Instead, Morley describes dynamically generating semantic routines on demand during emulation rather than statically storing all routines in the body of a software emulation system. Morley, col. 2, ll. 22-26. An application program consists of commands that are designed for the instruction set of an emulated processor, and a set of instructions in the native code which emulates instructions in emulated code is referred to as a semantic routine. Morley, col. 3, ll. 20-48. For the specific emulated instruction that corresponds to a semantic routine that is statically stored in emulator code, a dispatch table entry comprises a pointer to a stored semantic routine. Morley, col. 2, .ll 20-37. In other words, Morley describes replacing pointers to each routine of an emulation system to semantic routines. Morley does not teach:

> **generating decoder source code**, prior to a simulation time, by ...identifying a plurality of unique instruction patterns across a plurality of input applications; selecting an appropriate template... configured to implement a functionality of an instruction contained within an instruction class...generating a customized template for each instruction of the plurality of instructions using parameter values contained in the instruction…**compiling the decoder source code to generate an optimized decoder having optimized decoded instructions**.

Claims 28, 32, and 40, emphasis added. Zemach does not provide such a teaching either. Zemach describes translating a sequence of target instructions on a host machine using a simulator. Zemach, col. 1, ll. 35-40. The simulator includes a binary translator to translate the target code into host machine code. Zemach, col. 1, ll. 41-48. The binary translator translates a sequence of target instructions and stores the translated code in a translation cache. Zemach, col. 1, ll. 56-62. Once translated, a block of translated code may be executed on the host processor a number of times from the translated cache. *Id.* Zemach does not teach:

> **generating decoder source code**, prior to a simulation time, by ...identifying a plurality of unique instruction patterns across a plurality of input

> applications; selecting an appropriate template… configured to implement a functionality of an instruction contained within an instruction class…generating a customized template for each instruction of the plurality of instructions using parameter values contained in the instruction…**compiling the decoder source code to generate an optimized decoder having optimized decoded instructions**.

Claims 28, 32, and 40, emphasis added. Because neither Leupers, Morley, nor Zemach, alone or in combination, teach or suggest all of the features of independent Claims 28, 32, and 40, Applicants respectfully submit that Claims 28-42 are patentable under 35 U.S.C §103(a) over Leupers, Morley, and Zemach.

B. DeWitt.

Applicants respectfully submit that new Claims 28-42 are patentable under 35 U.S.C. §103(a) over Leupers in view of Morley, further in view of Zemach, and further in view of DeWitt for at least the following reasons.

Independent Claims 28, 32, and 40 each recite:

> **generating decoder source code**, prior to a simulation time, by …identifying a plurality of unique instruction patterns across a plurality of input applications; selecting an appropriate template… configured to implement a functionality of an instruction contained within an instruction class…generating a customized template for each instruction of the plurality of instructions using parameter values contained in the instruction…**compiling the decoder source code to generate an optimized decoder having optimized decoded instructions**.

Claims 28, 32, and 40, emphasis added. For the reasons discussed above, neither Leupers, Morley, nor Zemach provides such a teaching. DeWitt does not provide such a teaching either. Instead, DeWitt describes obtaining performance data in a data processing system. DeWitt, ¶[0003]. As a processor processes instructions, an instruction cache determines which instructions are associated with performance indicators. DeWitt, ¶[0074]. Signals associated with performance indicators are sent to a performance monitor unit. *Id.* DeWitt does not teach:

> **generating decoder source code**, prior to a simulation time, by …identifying a plurality of unique instruction patterns across a plurality of input

> applications; selecting an appropriate template... configured to implement a functionality of an instruction contained within an instruction class...generating a customized template for each instruction of the plurality of instructions using parameter values contained in the instruction...**compiling the decoder source code to generate an optimized decoder having optimized decoded instructions.**

Claims 28, 32, and 40, emphasis added. Because neither Leupers, Morley, Zemach, nor DeWitt, alone or in combination, teach or suggest all of the features of independent Claims 28, 32, and 40, Applicants respectfully submit that Claims 28-42 are patentable under 35 U.S.C. §103(a) over Leupers, Morley, Zemach, and DeWitt.

C. Killian.

Applicants respectfully submit that new Claims 28-42 are patentable under 35 U.S.C. §103(a) over Killian, in view of DeWitt, further in view of Leupers, further in view of Morley, and further in view of Zemach for at least the following reasons.

Independent Claims 28, 32, and 40 each recite:

> **generating decoder source code**, prior to a simulation time, by ...identifying a plurality of unique instruction patterns across a plurality of input applications; selecting an appropriate template... configured to implement a functionality of an instruction contained within an instruction class...generating a customized template for each instruction of the plurality of instructions using parameter values contained in the instruction...**compiling the decoder source code to generate an optimized decoder having optimized decoded instructions**.

Claims 28, 32, and 40, emphasis added. For the reasons discussed above, neither DeWitt, Leupers, Morley, nor Zemach provides such a teaching. Killian does not provide such a teaching either. Instead, Killian describes automatically configuring a processor by generating a description of a hardware implementation of the processor and a set of software development tools for programming the processor. Killian, col. 6, ll. 32-37. A configured definition of a target instruction set is developed from a standardized language. Killian, col. 6, ll. 50-65. Instructions are defined into a class, and each instruction in a class has the same format and operand usage. Killian, col. 16, ll. 14-20. An instruction semantic statement describes the behavior of one or more instructions. Killian, col. 16, ll. 40-50. Killian does not teach:

> **generating decoder source code**, prior to a simulation time, by ...identifying a plurality of unique instruction patterns across a plurality of input applications; selecting an appropriate template... configured to implement a functionality of an instruction contained within an instruction class...generating a customized template for each instruction of the plurality of instructions using parameter values contained in the instruction...**compiling the decoder source code to generate an optimized decoder having optimized decoded instructions.**

Claims 28, 32, and 40, emphasis added. Because neither Killian,DeWitt, Leupers, Morley, Zemach, nor DeWitt, alone or in combination, teach or suggest all of the features of independent Claims 28, 32, and 40, Applicants respectfully submit that Claims 28-42 are patentable under 35 U.S.C. §103(a) over Killian, DeWitt, Leupers, Morley, and Zemach.

D. Nohl.

Applicants respectfully submit that new Claims 28-42 are patentable under 35 U.S.C. §103(a) over Killian, in view of DeWitt, further in view of Leupers, further in view of Morley, further in view of Zemach, and further in view of Nohl for at least the following reasons.

Independent Claims 28, 32, and 40 each recite:

> **generating decoder source code**, prior to a simulation time, by ...identifying a plurality of unique instruction patterns across a plurality of input applications; selecting an appropriate template... configured to implement a functionality of an instruction contained within an instruction class...generating a customized template for each instruction of the plurality of instructions using parameter values contained in the instruction...**compiling the decoder source code to generate an optimized decoder having optimized decoded instructions**.

Claims 28, 32, and 40, emphasis added. For the reasons discussed above, neither Killian, DeWitt, Leupers, Morley, nor Zemach provides such a teaching. Nohl does not provide such a teaching either. Instead, Nohl describes simulating a program by accessing a table of compiled instructions. Nohl, ¶[0019]. If compiled data for an instruction is not stored in the table, the instruction is compiled and data is stored in the table for it. *Id.* By storing compiled data in the table, the instruction need not be compiled again if the instruction is re-executed. *Id.* Compiled data includes information suitable for the simulator to execute the instruction. Norl, ¶[0039]. Nohl does not teach:

> **generating decoder source code**, prior to a simulation time, by …identifying a plurality of unique instruction patterns across a plurality of input applications; selecting an appropriate template… configured to implement a functionality of an instruction contained within an instruction class…generating a customized template for each instruction of the plurality of instructions using parameter values contained in the instruction…**compiling the decoder source code to generate an optimized decoder having optimized decoded instructions**.

Claims 28, 32, and 40, emphasis added. Because neither Killian, DeWitt, Leupers, Morley, Zemach, nor Nohl, alone or in combination, teach or suggest all of the features of independent Claims 28, 32, and 40, Applicants respectfully submit that Claims 28-42 are patentable under 35 U.S.C. §103(a) over Killian, DeWitt, Leupers, Morley, Zemach, and Nohl.

E. Wang.

Applicants respectfully submit that new Claims 28-42 are patentable under 35 U.S.C. §103(a) over Killian, in view of DeWitt, further in view of Leupers, further in view of Morley, further in view of Zemach, and further in view of Wang for at least the following reasons.

Independent Claims 28, 32, and 40 each recite:

> **generating decoder source code**, prior to a simulation time, by …identifying a plurality of unique instruction patterns across a plurality of input applications; selecting an appropriate template… configured to implement a functionality of an instruction contained within an instruction class…generating a customized template for each instruction of the plurality of instructions using parameter values contained in the instruction…**compiling the decoder source code to generate an optimized decoder having optimized decoded instructions**.

Claims 28, 32, and 40, emphasis added. For the reasons discussed above, neither Killian, DeWitt, Leupers, Morley, nor Zemach provides such a teaching. Wang does not provide such a teaching either. Instead, Wang describes adding advanced instructions to a microprocessor. Wang, ¶[0008]. A processor generated using instruction extensions includes an instruction fetch unit to decode the advanced instructions added. Wang, ¶[0031]. Operation classes associate opcodes with operands, and opcodes determine the behavior of an instruction including the opcode. Wang, ¶[0102]. Opcodes are used to determine hardware and software associated with an execution unit for the opcode. *Id.* Wang does not teach:

> **generating decoder source code**, prior to a simulation time, by ...identifying a plurality of unique instruction patterns across a plurality of input applications; selecting an appropriate template... configured to implement a functionality of an instruction contained within an instruction class...generating a customized template for each instruction of the plurality of instructions using parameter values contained in the instruction...**compiling the decoder source code to generate an optimized decoder having optimized decoded instructions**.

Claims 28, 32, and 40, emphasis added. Because neither Killian, DeWitt, Leupers, Morley, Zemach, nor Wang, alone or in combination, teach or suggest all of the features of independent Claims 28, 32, and 40, Applicants respectfully submit that Claims 28-42 are patentable under 35 U.S.C. §103(a) over Killian, DeWitt, Leupers, Morley, Zemach, and Wang.

Thus, the cited combination of references does not disclose, nor suggest, the novel features of the inventions claimed in claims 28-42.

III. Conclusion

Prompt and favorable action on the merits of the claims is earnestly solicited. Should the Examiner have any questions or comments, the undersigned can be reached at (949) 567-6700.

The Commissioner is authorized to charge any fee which may be required in connection with this Amendment to deposit account No. 15-0665.

Respectfully submitted,
ORRICK, HERRINGTON & SUTCLIFFE LLP

Dated: 02/21/2012

By:/Kenneth S. Roberts/
Kenneth S. Roberts
Reg. No. 38,283

Orrick, Herrington & Sutcliffe LLP
2050 Main Street, Suite 1100
Irvine, CA 92614-8255
Tel. 949-567-6700
Fax: 949-567-6710